

FLOW3/TYPO3 5.0: A status report

Andreas Förthner
<andreas.foerthner@netlogix.de>

Overview

- ☛ About me
- ☛ TYPO3 5.0 vs. FLOW3
- ☛ Developing the FLOW3 way
- ☛ Important Components for FLOW3
- ☛ Latest implementations
- ☛ Go to TYPO3 5.0
- ☛ Questions

About me

- ♥ Andreas Förthner, born 04.12.1985
- ♥ Working with TYP03 in customer projects for netlogix in Nuremberg, Germany since 2003
- ♥ Studying computer science in Erlangen, Germany
- ♥ Member of the TYP03 5.0/FLOW3 Core Team since summer 2007

Why do we need TYP03 5.0?

- ♥ TYP03 v4 is (mostly?) feature complete
- ♥ Latest releases: focus on usability
- ♥ Grown architecture (no Unit-Tests!)
- ♥ Bigger changes are risky (impossible?)
- ♥ TYP03 is often used as application framework – but it's a CMS

OK 5.0, but why FLOW3?

- ♥ Many planned features are not only useful for a CMS
- ♥ Why not share these things with the whole PHP community?
- ♥ Infrastructure should be decoupled from CMS specific things

What exactly is FLOW3?

- It's a...
- Shipped...
- Pack...
- PHP...
- By default uses a JSR-283-based Content Repository for persisting data

**It's the basis for
TYP03 5.0!**

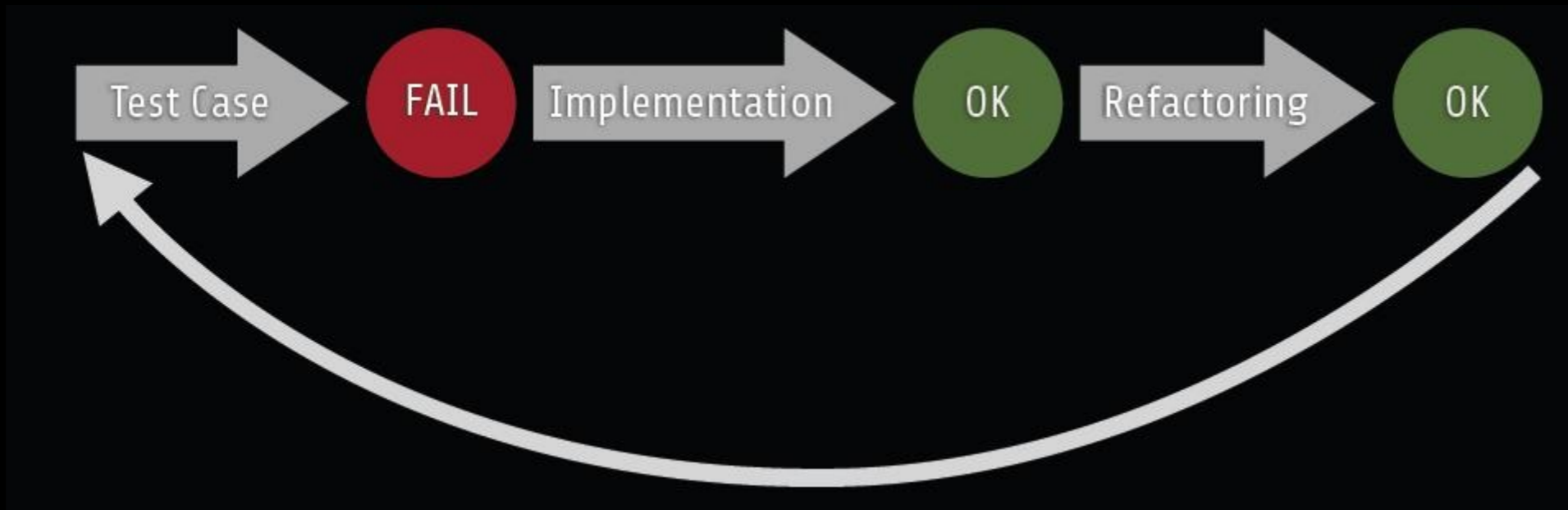
Developing the FLOW3 way

Get into the flow...

- ♥ Test Driven Development (TDD)
- ♥ Dependency Injection (DI)
- ♥ Aspect Oriented Programming (AOP)
- ♥ Domain Driven Design (DDD)

Developing the FLOW3 way

Test Driven Development



Swiss TYP03 Usergroup Conference 2009

FLOW3/TYP03 5.0: A status report

Inspiring people to
share

Developing the FLOW3 way

Dependency Injection

- ♥ Hollywood Principle: “Don't call us, we'll call you”
- ♥ Using the Inversion of Control (IoC) pattern, to create and initialize objects

Developing the FLOW3 way

Dependency Injection

```
class LoginService {  
    protected $session;  
  
    protected $logger;  
  
    public function __construct() {  
        $this->session = new SomeSessionImplementation();  
        $this->logger = new SomeLoggerImplementation();  
    }  
}
```

We told you:
Don't call us!

Developing the FLOW3 way

Dependency Injection

```
class LoginService {  
  
    protected $session;  
  
    protected $logger;  
  
    public function __construct(SessionInterface $session,  
                                LoggerInterface $logger) {  
        $this->session = $session;  
        $this->logger = $logger;  
    }  
}
```

Developing the FLOW3 way

Aspect Oriented Programming

- ♥ AOP is a programming paradigm
- ♥ Complements OOP
- ♥ OOP: group concerns in classes
- ♥ AOP: crosscutting concerns (concerns belonging to many classes, e.g. security or persistence)

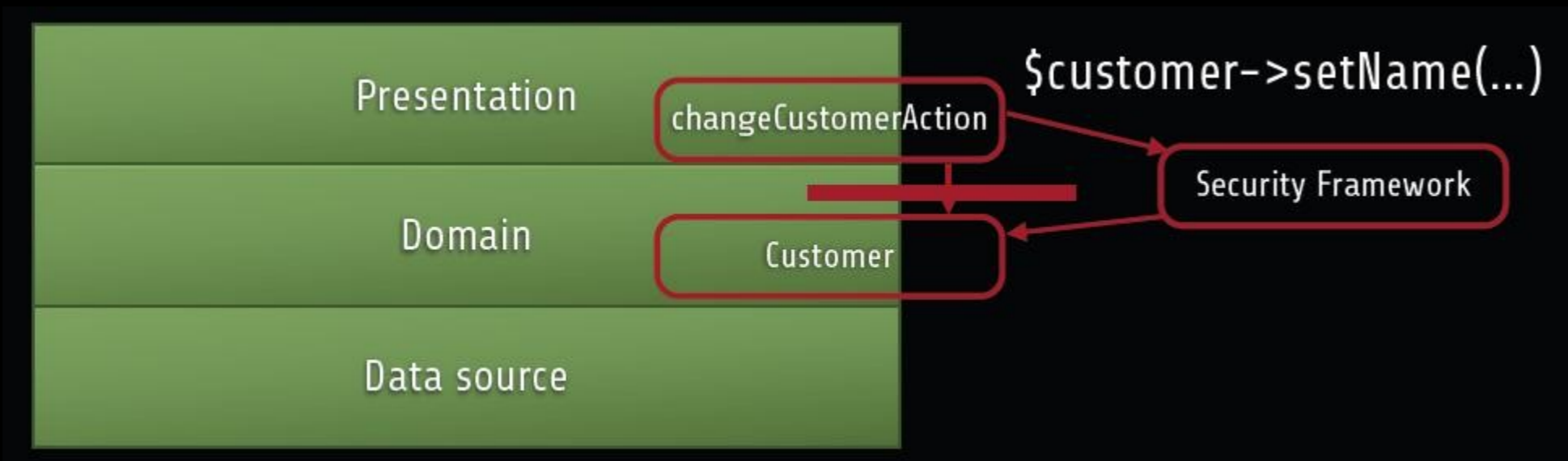
Developing the FLOW3 way

Domain Driven Design

- ♥ Domain = The things you want to get done
- ♥ Develop only the things of your domain
- ♥ Create a common language: every class, method, variable should be understandable in the context of your domain
- ♥ Infrastructure like, persistence, security, request-response handling is not in your domain

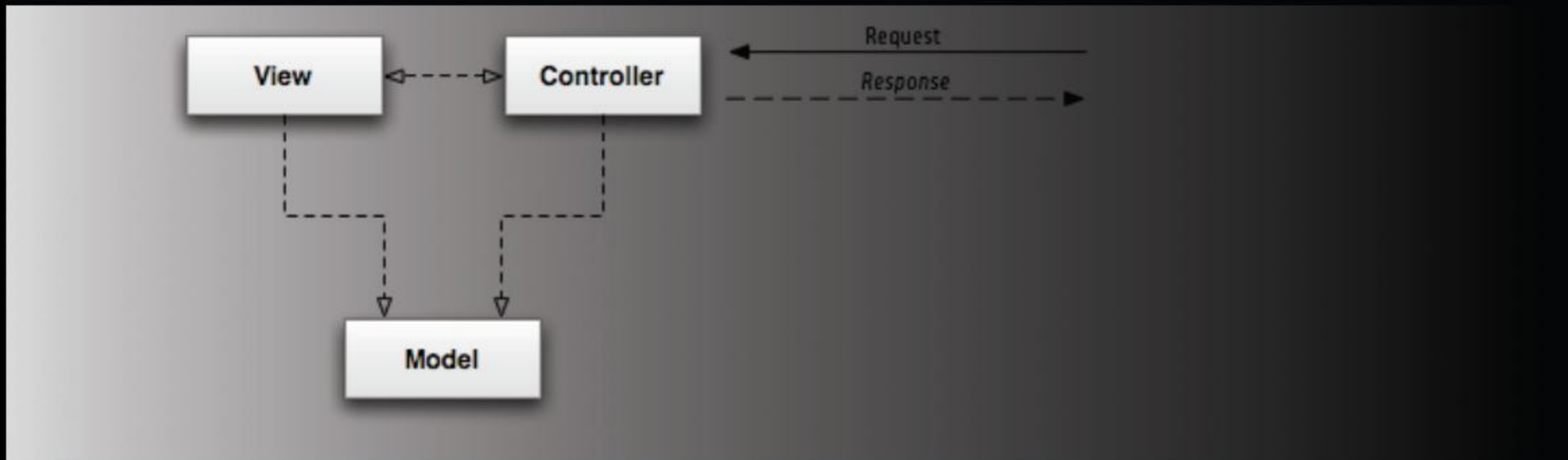
Developing the FLOW3 way

Aspect Oriented Programming



Important Components for FLOW3

Model – View – Controller (MVC)



Swiss TYPO3 Usergroup Conference 2009

FLOW3/TYPO3 5.0: A status report

Inspiring people to
share

Important Components for FLOW3

MVC

- ♥ Call actions to do anything
- ♥ Different views for different presentations of your data
- ♥ Model: This is the data and operations of your domain (-> DDD)
- ♥ Arguments are filtered, validated, transformed according to the specified datatype

Important Components for FLOW3

Persistence

- ❖ You don't have to care about persistence
- ❖ Every persisted object has to be added to and can be retrieved from a repository
- ❖ Annotations are important!

Important Components for FLOW3

Security

- ♥ Like persistence you don't have to code it
- ♥ The framework will enforce security rules, without the need of changing code -> touchless!
- ♥ Every(!) method can be restricted declarative

Important Components for FLOW3

The “Fluid” Package

- ♥ A new template engine
- ♥ Problems of existing engines:
 - ♥ Not completely OOP
 - ♥ Hard to use for non-HTML templates
 - ♥ No autocompletion (!)
 - ♥ Not easy to extend

The “Fluid” Package

- ♥ Every tag has it's corresponding PHP class -> very easy to extend!

```
<f3:for each="{posts}" as="post">
    <h1><f3:format.date target="{post.date}" format="d-m-Y" />{post.title}</h1>
    <p>{post.content}</p>
    <f3:link action="show" arguments="{postUUID :
                                        post.identifier}">more...</f3:link>
</f3:for>
```

New things

Latest implementations

- ♥ Intelligent cache flushing
- ♥ File modification monitoring
- ♥ Clean unit tests
- ♥ Fluid
- ♥ RSA-Service

Go to TYPO3 5.0

Swiss TYPO3 Usergroup Conference 2009

FLOW3/TYPO3 5.0: A status report

Inspiring people to
share

Go to TYP03 5.0

The next weeks (until T3BOARD09?)

- ♥ Implement missing features and cleanup MVC
- ♥ Cleanup and add features for persistence and security
- ♥ Work on the TYP03 package: Robert will start again on monday
- ♥ Beginning of February: UI meeting with Rasmus, Jens and Robert -> work on BE design
- ♥ flow3.org will be built with TYP03 5.0

Go to TYP03 5.0

What happens to TYP03 4.x?

- ♥ See TYP03 Transition Days and it's central result.
The Berlin Manifesto:
 - ♥ TYP03 v4 continues to be actively developed
 - ♥ v4 development will continue after the release of v5
 - ♥ Future releases of v4 will see its features converge with those in TYP03 5.0
 - ♥ TYP03 5.0 will be the successor to TYP03 v4
 - ♥ Migration of content from TYP03 v4 to TYP03 v5 will be easily possible

Patrick Broens, Karsten Dambekalns, Dmitry Dulepov,
 Andreas Förthner, Oliver Hader, Martin Herr, Christian
 Jul Jensen, Thorsten Kahler, Steffen Kamper, Christian
 Kuhn, Sebastian Kurfürst, Martin Kutschker, Robert
 Lemke, Tobias Liebig, Benjamin Mack, Peter Niederlag,
 Jochen Rau, Ingo Renner, Ingmar Schlecht, Jeff
 Segars, Michael Stucki, Bastian Waidelich

Questions?

Swiss TYP03 Usergroup Conference 2009

FLOW3/TYP03 5.0: A status report

Inspiring people to
share

TYPO3 